A 2D Image Motion Detection Method Using a Stationary Camera

Luís A. Alexandre[1, 2], Aurélio C. Campilho[1, 3]

[1]*INEB - Instituto de Engenharia Biomédica*
Praça Coronel Pacheco,1, 4050 Porto, Portugal
[2]*UBI - Universidade da Beira Interior*
Rua Marquês d'Ávila e Bolama, 6200 Covilhã, Portugal
E-mail:lfbaa@alpha2.ubi.pt
[3]*FEUP - Faculdade de Engenharia da Universidade do Porto*
Rua dos Bragas , 4099 Porto Codex, Portugal
E-mail:campilho@garfield.fe.up.pt

Abstract: *This paper describes a 2D motion detection method developed for a road traffic monitoring system. Such systems collect data allowing the management of traffic, increasing road security and traffic capacity.*
The objective of the development of this method was to allow the construction of a road traffic monitoring system that would work in real time, based on low cost hardware, namely a video camera, an image acquisition board and a Pentium 133MHz personal computer.
In order to the system work in real time the algorithm for the 2D motion detection had to be simple and at the same time provide the desirable high vehicle detection rate.

Key Words: *motion detection, road traffic monitoring, real-time system.*

1. INTRODUCTION

At present there are several methods for motion detection that can be gathered in two broad classes: feature based and optical flow based. For the specific application of road traffic monitoring, where real time is mandatory and a static camera is used, there are two subclasses namely: interframe differencing and reference frame differencing methods.

Our approach is based on reference frame differencing. Following this approach there are several methods. The method proposed in [4] is based on the threshold of a difference image obtained by subtracting a reference frame to the current frame. The reference frame is refreshed choosing randomly 64 pixels that are substituted by the corresponding pixels in the current frame.

In [5] it is described a motion detection method used in a commercial system developed for vehicle counting for automatic toll billing. The method is based on the difference between the histogram of a reference frame and the histogram of the current frame. If the difference is considerable a vehicle is counted.

A motion detection method based on the analysis of the difference frame between the captured frames and a reference frame is proposed in [2]. This reference frame is build by an exponential temporal smoothing of the captured frames. The motion detection is made by comparing the intensity of the pixels in the difference frame with a threshold previously defined.

A motion detection technique that uses morphological edge detection (SMED - Separable Median Filter) and reference frame differencing is presented in [3]. This technique obtains the edges of the current and the reference frame and subtracts them obtaining the edges of the moving objects.

In section 2 we describe the motion detection method. In 2.1 the detection stage is described, where the initial processing is done. In 2.2 we describe the decision stage, where the final decision about a vehicle detection is produced, based on the a priori knowledge about the scene and on the output of the detection stage. In section 3 the reference line updating is described and in section 4 some considerations on the method's performance are presented. Finally, the conclusions are given in section 5.

2. THE 2D MOTION DETECTION METHOD

Since the objective of this method is real-time 2D motion detection on image sequences using low cost hardware, the amount of data to be processed must be small. It is not possible to process the 25 images per second of the video signal with full resolution (768 x 576) using low cost hardware (even if we only use 8 bits per pixel).

We have to do data reduction. However it is not possible to reduce the temporal dimension, by processing only some of the images captured since time resolution is essential for the accurate measure of vehicle velocities, in a traffic monitoring system. Our method uses spatial data reduction. Only two line segments for each sensor placed on the screen are processed. Figure 1 shows a “sensor”: it is composed of two “detection lines”. The system user places the detection lines on the screen. They can be moved to a different position if the scene changes.

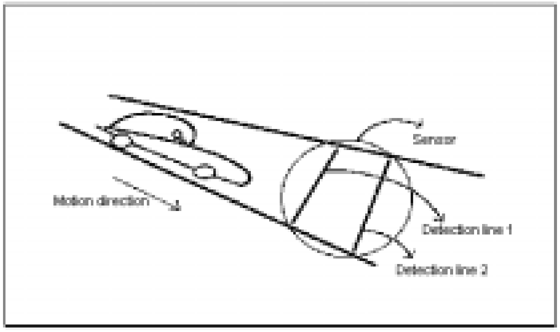

Figure 1. The traffic sensor, formed by two detection lines.

We can represent a detection line at a given instant t, by a vector $x(t)$, with n elements (the pixels that make up the line):

$$x(t) = [x_1, x_2, \ldots , x_n]^T$$

Each x_i can have a value between 0 and 255, since we use monochromatic images with 8 bits per pixel.

The method maintains a reference line $ref\_line(t)$, for each detection line $x(t)$, which is composed only by background information of the detection line (see 3).

The method has two stages: the first is the detection stage, and the second the decision stage.

2.1. The Detection Stage

This stage does not assume any a priori knowledge about the scene. It simply discriminates between a vector "similar" to the reference line and another that is not "similar" to the reference line. The criterion for similarity is given by a linear classifier. We will discuss this further bellow.

The method starts by subtracting the reference line from the respective detection line, giving a new vector $y(t)$:

$$y(t) = x(t) - ref\_line(t)$$

This step allows the independence of the results from the scene and from the location of the sensor in the scene.

The result $y(t)$ is then analysed through the calculation of two statistical measures: the variance (*var*) and a measure that approximates the spatial derivative of the image of the line segment (*sqd*).

The *var* is defined by

$$var(t) = \frac{1}{n-1}\sum_{i=1}^{n}\left(\bar{x} - x_i\right)^2$$

where

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

and

$$sqd(t) = \frac{1}{n-1}\sum_{i=1}^{n-1}\left(x_{i+1} - x_i\right)^2$$

These two features are organised in a vector $z(t)$ defined by

$$z(t) = [\, var, sqd\,]^T$$

In figure 2 we can see a plot of several $z(t)$ vectors. The dots represent vectors when a vehicle is not over a detection line (case 1) and the plus signs represent vectors when a vehicle is over a detection line (case 2).

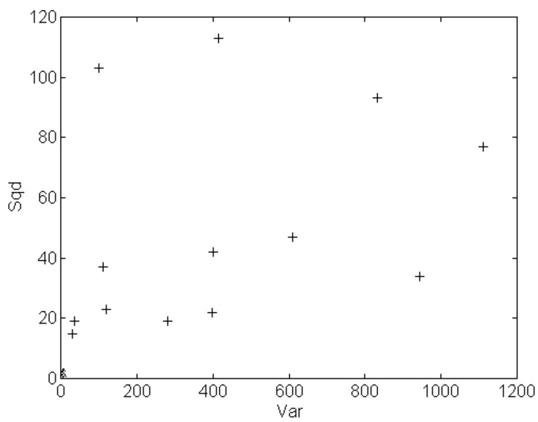

Figure 2. The dots near the origin represent detection lines without vehicles. The plus signs represent detection lines with vehicles.

As we remove the reference frame, the measures on *y(t)* should be very small when in the presence of a case 1, since if no vehicle is over the detection line it should be very similar to the reference line. In contrast, if in presence of a case 2, those measures can take very different values and can spread almost all of the *var x sqd* plane.

These two measures allow the system to discriminate between the presence or not of a vehicle on the detection line. This discrimination is based on the comparison of the measured quantities with predefined limits. These limits are obtained after the set-up of the system, automatically, by the analysis of some initial frames and the calculation of the same measures on those frames. These limits are then used to build a simple linear classifier, such as that on figure 3, which is used to automatically classify the vectors *z(t)*.

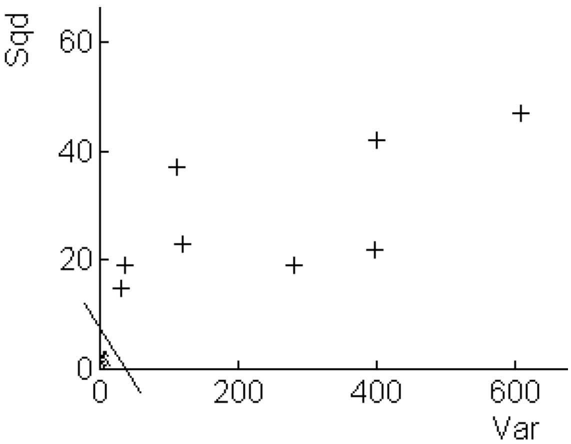

Figure 3. This is a section of figure 2. The line is a simple linear classifier used to separate the two populations.

2.2. The Decision Stage

In this stage the method processes the information produced by the detection stage. A decision is made based on the knowledge about the scene and about vehicle motion, of whether a vehicle has crossed the sensor or not.

To decide whether or not a vehicle has crossed the sensor at a time t_i the method uses the classification made by the detection stage of the two detection lines that compose the sensor, for the instants t_{i-1}, t_i and t_{i+1}. This way the decision stage incorporates a mechanism that is based on historical data of the system. Table 1 presents the conditions for sensor activation on instant t_i.

Instant	Line 1	Line 2
t_{i-1}	Vehicle	Road
t_i	Vehicle	Vehicle
t_{i+1}	Don't care	Vehicle

Table 1. Sensor activation conditions

The fact that the detection lines are ordered in the sensor (the first line a vehicle encounters during normal motion is line one) allows the method to ignore crossings that occur in the reverse order (notice that the columns in table 1 are not symmetric). These crossings might be caused by vehicles (that travel in the opposite direction) that are overtaking others and come out of their lanes. Another effect of this procedure is to detect only a vehicle which, at some time, is over both detection lines. This serves two purposes: the detection lines must be placed close enough so that the vehicles touches them both simultaneously and so the vehicle that touches the first line is the same one that touches the second line (avoids problems cause by overtaking and lane changing). Another advantage is that if the detection lines were somewhat separated from each other the speed measure would not be a good approximation to instantaneous speed.

3. REFERENCE LINE UPDATE

The reference lines are only updated when no vehicle has crossed that sensor for the last 3 frames. This way no information containing vehicles is incorporated on the reference frames. This allows us to give a large weight to each of the frames in the reference lines update. The update is made by averaging the existing reference line with the actual detection line, using the following expression

$$ref\_line(t+1) = 0,7 * ref\_line(t) + 0,3 * x(t)$$

Note that we give a high value (30%) to the weight of each new line. This value was determined empirically. Usually this expression uses a weight *k* that decreases exponentially with time [3],

$$ref\_line(t+1) = k * ref\_line(t) + (1-k) * x(t)$$

where $k = n/(n+1)$ and n indicates the number of the image (counted sequentially since the beginning of the update).

This large weight allows the system to react more promptly to sudden illumination variations such as those produced by clouds.

4. METHOD'S PERFORMANCE

This method has been tested in a traffic monitoring system and has produced high detection rates - 96,5% on 1487 vehicles [1].

There are situations where the method is unable to detect the presence of a vehicle. Some of these situations are represented in figures 4, 5 and 6.

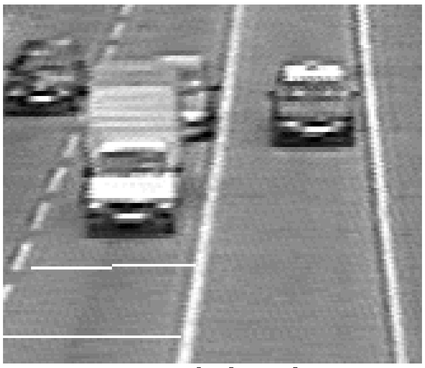

Figure 4. Vehicle occlusion.

Figure 5. Simultaneous sensor crossing.

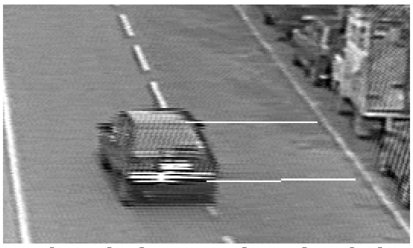

Figure 6. The vehicle is not aligned with the sensor.

Figure 4 shows a typical occlusion situation where one of the vehicles cover the other. For the system to distinguish the two vehicles a space between them, from the camera's point of view, had to exist. Figure 5 shows another situation where errors occur: the two vehicles crossed the sensor simultaneously and were counted as a single one. In figure 6, another situation is shown where sometimes false detections happen: the vehicle is not aligned with the sensor and only covers part of it. In these situations it is difficult to tell if a detection will occur, as it depends on the colour of the vehicle, its speed and the amount of the sensor that is covered.

These situations are responsible for almost all false detections. These situations have no solution from the method's point of view. It was designed in order to detect vehicles separated from each other by background (from the camera's viewpoint). It is also expected that only a vehicle at a time crosses the sensor and finally, the vehicles should cover a significant part of the sensor.

The remaining false detections are the ones that can be attributed to the method itself. One such situation is now discussed.

When the images are captured at night on a rainy day, the vehicle's headlights reflected on the road make a bright pattern in front of it that cause false detections. Some preliminary measurements showed that there can be as much as 50% false detections in this kind of situation (measurements made on the location shown in figure 7).

The problem also shows up in the opposite lane where these bright patterns sometimes cause the sensor to react and count a non-existent vehicle. A situation just as the one described is shown in figure 7.

To solve this problem an approach similar to [6] is being studied, where the type of illumination determines which detection parameters to use.

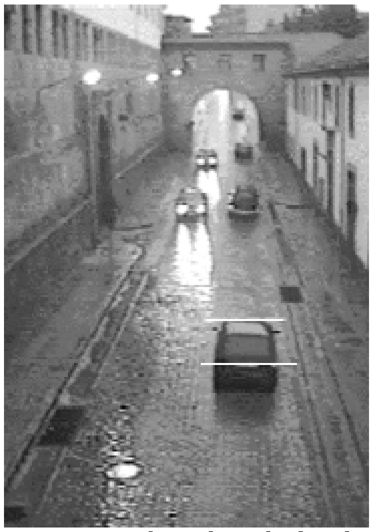

Figure 7. Situation where the vehicle's headlights cause some false detections.

5. CONCLUSION

We present a new method for 2D motion detection for traffic monitoring systems that is sufficiently fast to allow the implementation of such systems using low cost hardware. This method as been tested in a traffic monitoring system and as produced high detection rates - over 96%. There are still some situations where the method needs to be fine tuned (rainy days by night). These are presently being studied.

6. REFERENCES

[1] Luís A. Alexandre, Detecção de movimento e sua aplicação à monitorização de tráfego rodoviário, MSc. Thesis, FEUP, 1997.

[2] A.M.B. Cruz, Image Processing Applications Using A Transputer-Based System, PhD. Thesis, Univ. Bristol, 1995.

[3] M. Fathy, M.Y. Siyal, An image detection technique based on morphological edge detection and background differencing for real-time traffic analysis, Pattern Recognition Letters, 16, p. 1321-1330,1995.

[4] A. Houghton, et al, Automatic monitoring of vehicles at road junctions, Traffic Engineering and Control, vol 28, p. 541-543, 1987.

[5] D. Maravall, et al., Histogram difference based mobile detection. Application to automatic route vehicle counting, VI Spanish Symposium on PRIA, p. 535-542, 1996,

[6] Wixson, L., 1996, Illumination assessment for vision-based traffic monitoring, IEEE Proc. of ICPR 96, p. 56-62.